BLE and UART Bidirectional Transparent Transmission chip

CH9141

Datasheet

Version: 2A

http://wch.cn

1. Overview

The CH9141 is a chip realizing bidirectional transparent transmission between *Bluetooth*® low energy (BLE) and UART, supporting broadcast mode, host mode and slave mode, BLE4.2. It supports serial AT configuration and BLE communication configuration in slave mode and MODEM contact signal. The chip provides universal GPIO, synchronous GPIO and ADC acquisition functions, with the baud rate of UART up to 1Mbps. In the BLE slave mode, parameters such as BLE name and manufacturer information can be set, which can be easily configured through APP or serial port commands, which is convenient and fast.

The virtual serial port driver on the computer is provided to directly use serial port debugging tool for the BLE interface. It is compatible with the serial port application program, so it can communicate with the serial port without secondary development, easily realizing the exemption of wire plugging and not being limited by the cable distance.

The figures below are some application block diagrams of CH9141:

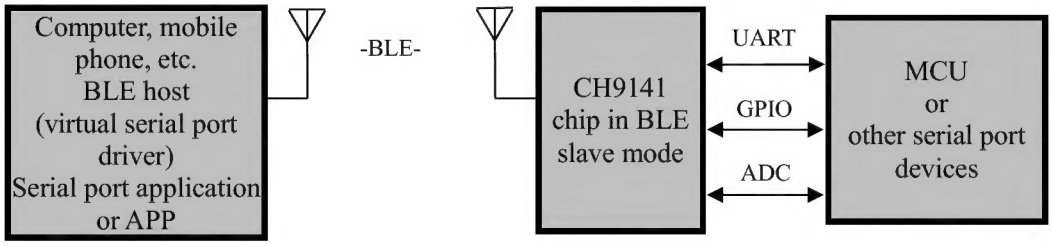


Figure 1 Block Diagram of Serial Communication between BLE Host and Serial Port Devices through CH9141

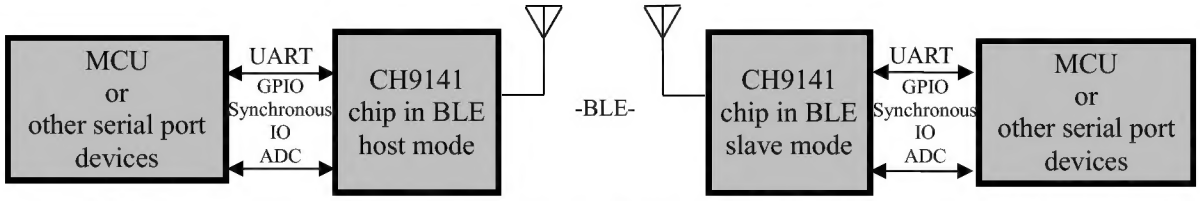


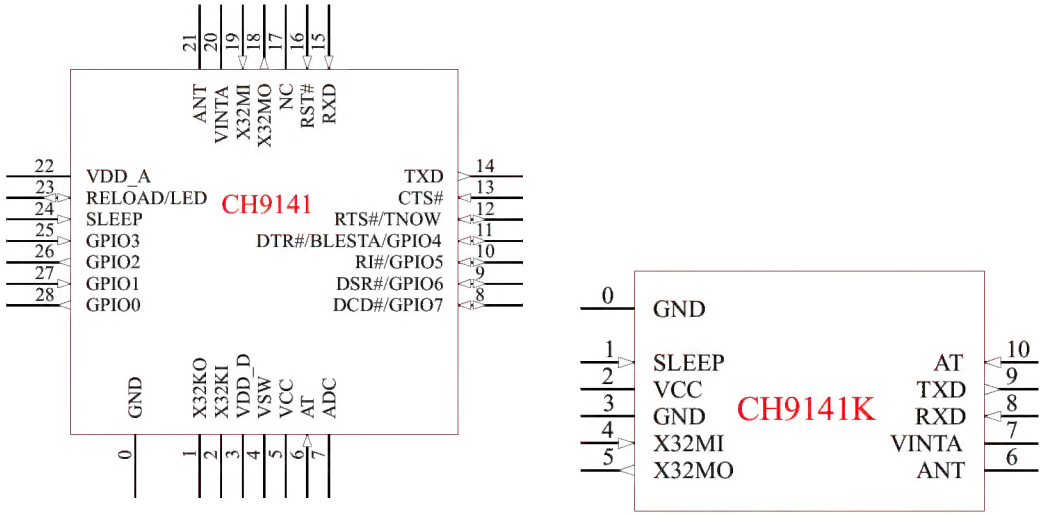
Figure 2 Block Diagram of communication between MCUs or Serial Port Devices on Both Ends through CH9141 in BLE Host-slave mode

2. Features

- Support broadcast mode, master mode and slave mode.
- Support serial port AT configuration and BLE transmission configuration.
- Provide computer-end BLE virtual serial port driver.
- Compatible with existing serial port software and tools, and no secondary development is required.
- Support BLE host connection for Windows/Linux/Android/iOS and other systems.
- Provide general GPIO and synchronous GPIO functions, support BLE control.

- Support one-channel 12-bit ADC acquisition and BLE reading.
- The power-off sleep current is 0.3uA.
- Transmission distance: 100 m
- Adjustable 8 positions of transmission power.
- Support 3.3V and 2.5V operating voltage.
- The maximum communication baud rate is 1Mbps.
- UARTs support the MODEM communication signals RTS, DTR, DCD, RI, DSR and CTS.
- UARTs support 5, 6, 7 or 8 data bits and 1 or 2 stop bits.
- UARTs support the parity check modes of odd, even, no check, blank 0 and mark 1.
- Support obtaining the chip supply voltage parameters.
- 32K clock source, optional external crystal oscillator.
- Provide QFN28 square leadless 28-pin package and ESSOP10 package.

3. Package



Model	Package	
	Name	Description
CH9141	QFN28	4*4mm
CH9141K	ESSOP10	6*5mm

4. Pins

QFN28 Pin No.	ESSOP10 Pin No.	Pin Name	Pin Type	Description
0	0	GND	P	Ground
1	None	X32K0	A	Inverted output of low-frequency crystal oscillator
2		X32KI	A	Inverted input of low-frequency crystal oscillator
3		VDD_D	P	Power, requires an external 2.2uF decoupling capacitor
4		VSW	P	Internal DC-DC switch output

5	2	VCC	P	Chip power input, requires an external 2.2uF decoupling capacitor
6	10	AT	I	AT transparent transmission function switching pin 0: AT mode 1: Transparent transmission mode
7	None	ADC	A	Analog ADC input pin
8		DCD# /GPIO7	I/O	DCD#: MODEM input signal of UART, carrier detect. GPIO7: General-purpose input and output.
9		DSR# /GPIO6	I/O	DSR#: MODEM input signal of UART, data device ready. GPIO6: General-purpose input and output.
10		RI# /GPIO5	I/O	RI#: MODEM input signal of UART, ring indicator. GPIO5:General-purpose input and output.
11		DTR# /BLESTA /GPIO4	I/O	DTR#: MODEM output signal of UART, data terminal ready. BLESTA: BLE connection status output. GPIO4: General-purpose input and output.
12		RTS# /TNOW	I/O	RTS#: MODEM communication output signal, request to send, active low. TNOW: RS485 transceiver switch control pin of UART.
13		CTS	I	MODEM communication input signal, clear to send, active low.
14	9	TXD	O	Serial port transmit pin
15	8	RXD	I	Serial port receive pin
16	None	RST#	I	Chip reset pin, active low.
17		NC	N	Invalid pin, suspended
18	5	X32MO	A	Inverted output of high-frequency crystal oscillator
19	4	X32MI	A	Inverted input of high-frequency crystal oscillator
20	7	VINTA	P	Power node of internal analog circuit, requires an external 2.2uF decoupling capacitor
21	6	ANT	A	RF signal input and output
22	None	VDD_A	P	Power input of internal analog circuit, requires an external0.1uF decoupling capacitor
23		RELOAD /LED	I/O	RELOAD: Restore factory settings pin, active low LED: Chip status indicator signal output
24	1	SLEEP	I	Low-energy control pin, active low, integrated pull-up resistor
25	None	GPIO3	I	Synchronous input IO
26		GPIO2	O	Synchronous output IO
27		GPIO1	I	Synchronous input IO
28		GPIO0	O	Synchronous output IO
None	3	GND	P	Ground

Note: P: power pin; A: analog pin; I: input pin; O: output pin; N: empty pin

The multiplexing function takes precedence.

5. Functional Specification

5.1 Broadcast Mode

The broadcast mode supports serial transparent transmission of data to the chip, which will automatically update the broadcast data or update the broadcast data in AT mode. It is recommended to use AT commands to send BLE data. The data is the original BLE broadcast data and the chip will not verify the validity of the data, so the data must conform to the BLE protocol specification. The broadcast data is updated at the interval of 100mS by default. If multiple packets are sent during the broadcast data update period, the last packet of data will be used as broadcast data. When the serial transparent transmission is used to update broadcast data, it is recommended to use serial port timeout mechanism. The interval between each packet must be greater than the UART timeout time. The broadcast data packet cannot exceed 30 bytes, and the format of the packet should refer to the BLE protocol specification.

5.2 Host Mode

The host mode only supports the CH914x slave chip of our company. This mode supports scan connection and direct MAC address connection. The device that scan echoes only displays the CH914x slave chip. After scanning, the results of the scan echo can be used for serial number connection. When connecting, input the serial number and slave connection password in AT mode. Or directly input the MAC address and password of the slave needed to be connected without scanning, and the chip will automatically connect to the device. The host mode supports set the MAC for automatic connection. If set, the host will automatically connect to the device after initialization. This method does not require scanning or other operations.

5.3 Slave Mode

In slave mode, the chip will transmit fixed broadcast data and support modify the chip name in the scan response data, that is, the chip name described below. The default broadcast interval is 100mS. The slave supports four basic BLE services, where the transparent transmission service UUID is 0xFFF0.

The UUIDs of the communication are 0xFFF1, 0xFFF2, and 0xFFF3. Please refer to the following table for instructions.

UUID	Property	Description
0xFFF1	Public Notice	The data received by UART will be transmitted to the host through this channel, and the host needs to enable notification. The data will be packaged in the size of MTU, and the exceeded data will be transmitted by the chip in separate package.
0xFFF2	Write only	Host transmit data channel, data will be sent out through UART.
0xFFF3	Read, write	Configuration channel, IO synchronization and other functions.

5.4 Serial Transparent Transmission Function

The serial transparent transmission uses a UART, and the default factory baud rate is 115200bit/s. The UART receive buffer of the chip is 512 bytes, and the serial port will transmit data by BLE in real time while receiving data. It is recommended to ensure the end of data transmission when performing AT configuration. When entering AT configuration, the currently received transparent transmission data will be lost if it is not saved, and data that has been saved in the receiving buffer will continue to be sent when exiting the AT mode. The data received by the BLE will be sent directly to the UART, and the data will not be temporarily stored. When there is a lot of BLE data, it will wait to be sent. If it has entered the AT mode currently, the data

received by the BLE will be directly discarded. It is recommended that the host make some speed restrictions when sending to reduce packet loss and buffer overflow.

Because BLE communication rate is related to its environment, it is recommended to use CTS/RTS flow control to prevent buffer overflow when the serial baud rate exceeds 9600bit/s and average RSSI of BLE is less than -70dBm.

5.5 Parameter Configuration

There are two configuration methods: one is to configure and control via BLE in the device mode, and the other is to enter the AT configuration mode through the serial port for configuration.

The BLE configuration is only used in the slave mode and cannot be used in the broadcast mode and the host mode. The configuration commands can be operated using the interface in the routine. After configuration, the chip will automatically save the parameters that take effect at the next startup.

The serial port configuration means configuration performed when the serial port enters AT mode. When the serial port performs AT operation, it must wait until the chip responds before sending the next data packet, and it cannot send continuously. The data is uniformly transmitted in ASCII characters, and the parameters in HEX format are also converted into ASCII format.

When the configuration parameters are modified, some need to be restarted to take effect. It is recommended to restart the chip after all the parameters are modified.

5.6 Device Information

The device information properties supported by the slave mode include: System ID, Model Number String, Serial Number String, Firmware Revision String, Hardware Revision String, Software Revision String, Manufacturer Name String, PnP ID. These parameters can be modified by the user. The String format supports string input and the size does not exceed 18 characters. In addition, the System ID and PnP ID parameters need to comply with the BLE protocol. If the System ID parameter is not configured by the user, the chip will automatically generate one. If the PnP ID is not configured, it will be displayed with all 0s, which can be referred to in the following table.

UUID	Name	Description
0x2A23	System ID	System ID, 8 bytes
0x2A24	Model Number String	Chip name string
0x2A25	Serial Number String	Serial number string
0x2A26	Firmware Revision String	Firmware version information
0x2A27	Hardware Revision String	Hardware version information
0x2A28	Software Revision String	Software version information
0x2A29	Manufacturer Name String	Manufacturer name information
0x2A50	PnP ID	PnP ID, 7 bytes

5.7 Functional Pin

5.7.1 MODEM Contact Signal

CH9141 supports the commonly used MODEM contact signals RTS, DTR, DCD, RI, DSR and CTS. When the "Notification Service" of the BLE configuration channel is turned on, the status change of the MODEM will notify the host in the form of a command frame. Refer to Chapter 8 for details. The DTR and RTS output signals can be controlled by AT commands or sending control commands to the BLE configuration

channel. When the flow control function is enabled, the RTS pin will be automatically controlled by the chip.

5.7.2 RS485 Switch Pin TNOW

The RTS pin of the chip can be set as the output pin of the TNOW function and supports configuration via software and hardware. Software configuration: send configuration commands through AT commands or BLE APP to set the TNOW pin function on and polarity control. Hardware configuration: if it is not configured by software, RTS#/TNOW is set to the pull-up input mode after the chip is powered on, which is defaulted to high level. A 4.7K pull-down resistor can be connected externally, which is set to low level. After the chip is powered on, the functions of the RTS# and TNOW pins will be selected according to the input level. The TNOW pin outputs the high level by default when the serial port transmits data and outputs the low level when the serial port does not transmit data.

5.7.3 BLE Connection Status Pin BLESTA

The DTR pin of the chip can be set as the output pin of the BLESTA BLE connection status and supports software and hardware configuration. Software configuration: send configuration commands through AT commands or BLE APP to set the BLESTA pin function on and polarity control. Hardware configuration: if functions of GPIO and BLESTA are not configured by software, DTR#/BLESTA/GPIO4 is set to the pull-up input mode after the chip is powered on, which is defaulted to high level. A 4.7K pull-down resistor can be connected externally, which is set to low level. After the chip is powered on, the functions of the DTR# and BLESTA pins will be selected according to the input level. The BLESTA pin outputs low level by default when BLE is not connected, and outputs high level after BLE is connected.

The BLESTA pin function is only available in BLE host and slave mode.

5.7.4 GPIO Function

There are two types of GPIO functions: one is synchronous IO, which is used for synchronization between host chip and slave chip after connection; the other is general-purpose IO, which can be configured through AT commands or configured and debugged through configuration interface. When the functional pins of the chip work in other modes and the general-purpose GPIO is operated, the chip will automatically disable other functions of the GPIO, enable the GPIO function, modify and save the parameters, and use it as GPIO function automatically after the next power on or restart.

The chip provides 8 GPIOs, which are divided into 4 synchronous IOs and 4 general-purpose IOs. The serial numbers of GPIOs are 0-7, namely GPIO0, GPIO1...GPIO7.

Synchronous IOs include GPIO0, GPIO1, GPIO2 and GPIO3, of which GPIO1 and GPIO3 are inputs, and GPIO0 and GPIO2 are outputs. The input and output characteristics of synchronous GPIOs cannot be modified. When the host and slave are connected, the two chips will synchronize these 4 IOs. The synchronization rule is that the GPIO1 and GPIO3 inputs of the host correspond to the GPIO0 and GPIO2 outputs of the slave, while the GPIO1 and GPIO3 inputs of the slave correspond to the GPIO0 and GPIO2 outputs of the host. The shortest synchronization time is a connection event. It should be noted that: synchronous IOs will be interrupted by the serial port transparent transmission during synchronization. The data transmitted through the serial port takes precedence in BLE transmission. IO synchronization information will be transmitted only when it is free. Therefore, the synchronization of the synchronous IOs will be delayed during the serial port transparent transmission.

The general-purpose IOs include GPIO4, GPIO5, GPIO6, GPIO7. Each IO can be configured as input or output independently. IOs can be set and controlled or the IO level can be read through AT configuration,

and it can also be configured and controlled by sending commands through BLE configuration interface when in BLE slave operating mode. In terms of configuration, it is recommended not to modify the IO configuration midway. After each IO is set, the parameters will be saved once, so the IOs need to be fixed and configured once. Sending AT commands should be sent according to the input and output characteristics, that is, the IOs in input mode can only send read commands, and the IOs in output mode can only send setting commands.

5.8 ADC Function

For ADC function, the chip provides one-channel 12-bit ADC sampling. The sampling value of ADC can be read through AT commands or through the BLE configuration interface in device mode. The sampling input voltage range is 0-2V.

The chip can get the current supply voltage through the command codes to facilitate voltage monitoring of the battery-powered environment.

5.9 Low Energy Function

The low energy mode is divided into two sleep modes. One is sleep mode, BLE will continue to work and other unrelated peripherals will be turned off. The other is that chip enters the power-down mode, it directly resets and restarts when it wakes up. The sleep mode can be set by AT commands. It is recommended to select the corresponding sleep mode according to your own needs and do not change the sleep mode frequently.

In sleep mode, the master cannot send serial data, and the chip can be controlled to enter low energy mode by pulling down the low energy sleep control pin (SLEEP pin). This mode is not a real sleep mode, it automatically wakes up and processes BLE according to the request of BLE communication. The data received on the BLE will still be sent to the master through the serial port, and the BLE will automatically sleep if there is no transaction to process. However, the SLEEP pin must be pulled up when the master wants to send data, it should wait for more than 20mS and then send data. Otherwise it will cause data error. After the pin is pulled up, the chip will exit the low energy mode.

In power-down mode, all unrelated peripherals will be turned off, the BLE will stop running. The sleep pin (SLEEP pin) can be controlled, it can be pulled down to enter the power-down mode, and pulled up and waken up to reset the chip.

5.10 Restore Factory Settings

The chip provides a restore factory settings pin (RELOAD/LED). After the chip is powered on, it will detect the restore factory settings pin. If the pin is detected to be pulled down for 2 seconds continuously, the chip will be restored to the factory settings, and then the chip will operate with the factory setting parameters. After RELOAD detects and processes, the pin will be multiplexed as LED output pin.

5.11 LED Pin Function

The LED pin of the chip displays the current operating status of the module, which is active at low level and is defaulted to high level. An LED can also be connected to visually see the working status of the module.

In the broadcast mode, the LED is at high level by default. When receiving a command to update the broadcast data, it will flash once at a time of 50mS low and 50mS high.

In slave mode, the LED pin will enter the broadcast state after module initialization. At this time, the LED

will flash continuously at a time of 500mS low and 500mS high. When the host is connected, the LED pin will be pulled down to keep the external LED always on. At this time, the LED will flash once at a time of 50mS high and 50mS low if there is data transmission and reception, , and finally keep the low level state. After disconnected, the pin will re-enter the broadcast state.

In the host mode, the LED is at high level by default. When the scan is executed, the LED will flash continuously at 165mS low and 165mS high. The LED will be at high level and keep the default status until the scan is over. When the connection is executed, the LED pin will be pulled down after successful connection and the external LED will be kept on. When there is data transmission and reception, it will flash once at a time of 50mS high and 50mS low. If it is disconnected, the LED will be pulled up. At this time, if there is no AT command, the host module will actively reconnect to the slave. If the connection is successful, it will become connected and the pin will be pulled down.

When the low energy mode is enabled, the LED pin will output a constant high, and the LED status will not be updated.

5.12 External Crystal Oscillator

The operation of the chip requires an external 32M crystal oscillator, and the chip pin has a built-in load capacitor, so there is no need for an external load capacitor. If there are strict requirements on the frequency offset, it is recommended to reserve a location for the external load capacitor to facilitate testing and adjusting the frequency offset on the overall system.

The 32K low frequency crystal oscillator of the chip is not necessary. If an external 32K crystal oscillator is used, the chip will automatically switch to use the external crystal oscillator, and the pin has a built-in load capacitor, so there is no need for an external load capacitor. If the chip is working in host mode, it is recommended to connect an external 32Kcrystal oscillator to mention the accuracy of the clock.

5.13 Internal LSI and RF Calibration

The chip supports to calibrate the internal LSI clock and RF through timing and temperature difference. When using an external 32K crystal oscillator, there is no need to care about the internal 32K calibration setting and the LSI clock will not be calibrated. It is recommended to set a reasonable timing or temperature difference method to calibrate the chip in applications where the temperature difference is relatively large or the ambient temperature changes quickly.

Timing mode: Internal LSI clock calibration and RF calibration are performed regularly according to the time parameters that are set.

Temperature difference mode: According to the time parameters that are set, the temperature difference is timed to be sampled. When the temperature difference reaches the set threshold, internal LSI clock calibration and RF calibration will be carried out.

6. AT Instruction Set

6.1 Basic AT Format

In AT mode, the chip is as a serial slave device, and the master (host) connected to the chip is as a serial master device. AT commands are sent by the host and the chip responds.

The basic format of the command sent by host is:

<AT><+><command code><operator><parameter><{CR}{LF}>

Note: The basic format consists of command codes mostly, and some commands are different. For details, see the command set below. Where {CR}{LF} corresponds to the "\r" "\n" defined by the character format, the hexadecimal is: 0x0D and 0x0A, which are the carriage return and line break in ASCII. The {CR}{LF } in the command is used as a separator and terminator.

The basic format returned by the chip is:

Return parameter format: <parameter><{CR}{LF}><OK><{CR}{LF}>

Correct status return: <OK><{CR}{LF}>

Error status return: <{CR}{LF}><ERR:><error code><{CR}{LF}>

Note: The error code is a HEX form consisting of two ASCII characters. For example, the error code is the character "01", which means hexadecimal 0x01. The current error code and its meaning are shown in the figure below:

Error Code	Meaning
01	Cache error: The current chip does not have a cache to respond, you can try again later.
02	Parameter error: Some parameters of the AT command that is sent do not meet the specifications. Note that the chip will not judge all parameters and external guarantees for basic correctness is required.
03	Command not supported: Commands are not supported in the current mode, such as sending connection commands in broadcast mode, etc.
04	Command cannot be executed: The command cannot be executed temporarily, you can try again later. Generally because there is not enough cache to process this command and the chip is busy.

6.2 UART Configuration Command Set

No.	Command	Description
1	AT...	Enter AT configuration
2	AT+RESET	Reset chip
3	AT+VER	Get the chip version number
4	AT+HELLO	Inquire/set welcome note
5	AT+RELOAD	Reset all parameters
6	AT+SHOW	Display chip information
7	AT+SAVE	Save current parameters
8	AT+EXIT	Exit AT configuration
9	AT+GPIO	Inquire/set the general GPIO and synchronous GPIO
10	AT+INITIO	Set GPIO output initial value
11	AT+UART	Inquire/set UART parameters
12	AT+MAC	Inquire local MAC address
13	AT+TPL	Inquire/set transmission power
14	AT+BLESTA	Inquire BLE status
15	AT+DISCONN	Disconnect the current connection
16	AT+BLEMODE	Inquire/set the BLE working mode
17	AT+CCADD	Inquire current MAC address
18	AT+NAME	Inquire/set chip name

19	AT+PNAME	Inquire/set device name
20	AT+PASEN	Inquire/set password enabled
21	AT+PASS	Inquire/set password
22	AT+SYSID	Inquire/set system ID of device information
23	AT+MODNAME	Inquire/set chip name of device information
24	AT+SERINUM	Inquire/set serial number of device information
25	AT+FIRMREV	Inquire/set firmware version of device information
26	AT+HARDREV	Inquire/set hardware version of device information
27	AT+SOFTREV	Inquire/set software version of device information
28	AT+MANUNAME	Inquire/set manufacturer name of device information
29	AT+PNPID	Inquire/set PNP ID of device information
30	AT+ADVEN	Inquire/set broadcast enabled
31	AT+ADVDAT	Inquire/set broadcast data
32	AT+LINK	Connect to the specified BLE device according to the serial number
33	AT+CONN	Connect directly according to the given BLE device parameters
34	AT+SCAN	Host scan command
35	AT+CONADD	Inquire/set default connection parameters
36	AT+CLRCONADD	Clear default connection parameters
37	AT+RSSI	Set read RSSI
38	AT+ADC	Read ADC value
39	AT+SLEEP	Set chip sleep mode
40	AT+BAT	Read supply voltage of chip
41	AT+BDSP	Host scan to show slave voltage
42	AT+BLECFGEN	BLE configuration interface switch
43	AT+BCCH	Set broadcast channel
44	AT+ADVINTER	Set broadcast interval
45	AT+CONNINTER	Set connection interval
46	AT+LSICALI	Set internal 32K clock calibration
47	AT+RFCALI	Set BLE RF calibration
48	AT+TNOW	Set TNOW pin function
49	AT+BSTA	Set BLE status pin
50	AT+AFEC	Set flow control and output pin
51	AT+IOEN	Set GPIO function enabled

1. Enter AT configuration

Command: AT...{CR}{LF}

Note: This command is used when the pin is not used to enter the AT configuration. It is sent when UART is free for 500mS, the chip will enter the AT mode and return the status. Sending the command after entering the AT mode will also give a response.

Return: OK{CR}{LF}

For example: Host sends: AT...{CR}{LF} Chip responds: OK{CR}{LF}

2. Reset chip

Command: AT+RESET{CR}{LF}

Note: This command is used to reset the chip after 20mS.

Return: OK{CR}{LF}

3. Get the chip version number

Command: AT+VER{CR}{LF}

Note: This command is used to get the chip version, and return the version number and status

Return: VER:<version number>{CR}{LF}OK{CR}{LF}

For example: Host sends: AT+VER{CR}{LF} Chip responds: VER:V1.00{CR}{LF}OK{CR}{LF}

4. Welcome note

Getting command: AT+HELLO?{CR}{LF}

Note: This command is used to get the current welcome note, which can be used as a sign of completing chip initialization.

Return: <welcome note string>{CR}{LF}OK{CR}{LF}

For example: Host sends: AT+HELLO?{CR}{LF} Chip responds: Welcome{CR}{LF}OK{CR}{LF}

Setting command: AT+HELLO=<welcome note string>{CR}{LF}

Note: This command is used to set the current welcome note. When the welcome not is not used, the string will be empty. And the number of the string must be less than 30.

Return: Setting status. If it is correct, the setting is successful. If it is wrong, the error code will be queried.

For example: Host sends: AT+HELLO=WCH MODULE{CR}{LF} Chip responds: OK{CR}{LF}

For example: Host sends: AT+HELLO={CR}{LF} Chip responds: OK{CR}{LF}, this function will not be enabled

5. Reset all parameters

Command: AT+RELOAD{CR}{LF}

Note: This command will reset all chip parameters, that is, restore factory settings and save.

Return: Status.

For example: Host sends: AT+RELOAD{CR}{LF} Chip responds: OK{CR}{LF}

6. Display information

Command: AT+SHOW{CR}{LF}

Note: This command is used to display some chip information.

Return: Information parameters and status.

For example: Host sends: AT+SHOW{CR}{LF} Chip responds: <chip information>OK{CR}{LF}

7. Save current parameters

Command: AT+SAVE{CR}{LF}

Note: This command is used to save all parameters currently set. The setting command will be saved automatically, and this command will conflict with the saved one. It is recommended not to use this command.

Return: Status.

For example: Host sends: AT+SAVE{CR}{LF} Chip responds: OK{CR}{LF}

8. Exit AT configuration

Command: AT+EXIT{CR}{LF}

Note: This command is used to exit the current AT configuration mode. It is used together with the enter configuration command. When AT pin is used to enter AT configuration mode, this command cannot be

used.

Return: OK{CR}{LF}

For example: Host sends: AT+EXIT{CR}{LF} Chip responds: OK{CR}{LF}

9. Set GPIO

Getting command: AT+GPIO<x>?{CR}{LF}

Note: x is 4-7 in the ASC format. Getting the pin level status of x will cause changes in GPIO settings. When the current x pin is set to input mode, it will directly return the IO level status that is read. When the current x pin is set to output mode, using this command will reconfigure the pin to input mode, save the parameters, and return the read IO level status. It is recommended that IO is fixed in one mode without modifying its input/output configuration midway.

Return: <level value: 0,1>{CR}{LF}OK{CR}{LF}

For example: Host sends: AT+GPIO4?{CR}{LF} Chip responds: 1{CR}{LF}OK{CR}{LF}

Setting command: AT+GPIO<x>=<level value: 0,1>{CR}{LF}

Note: Same with the acquisition, if the mode is wrong, the settings will be set and saved. It is also recommended not to modify the input and output configuration midway.

Return: OK{CR}{LF}

For example: Host sends: AT+GPIO4=0{CR}{LF} Chip responds: OK{CR}{LF}

10. Set GPIO output initial value

Getting command: AT+INITIO?{CR}{LF}

Note: This command is used to get the default value of GPIO output in the configuration. This value affects the value of IO output after the power-on initialization is completed, but the IO value of corresponding input state is not affected. The get value is a HEX converted ASCII, such as: 0xcc, which means binary 11001100, bit7-bit0 respectively correspond to GPIO7-GPIO0. Output low level when it is 0, and output high level when it is 1.

Return: Execution status.

For example: Host sends: AT+INITIO?{CR}{LF} Chip responds: CC{CR}{LF}OK{CR}{LF}

Setting command: AT+INITIO=<8 IO level values>{CR}{LF}

Note: The set parameter format is consistent with the get parameter format. Each bit indicates the corresponding IO power-on output status. After setting, the host will save the parameters.

Return: Execution status

For example: Host sends: AT+INITIO=00{CR}{LF}, all IOs output after power on are at low level Chip responds: OK{CR}{LF}

11. UART configuration

Getting command: AT+UART?{CR}{LF}

Note: This command returns the current UART parameter configuration.

Return: <baud rate>,<data bit>,<stop bit>,<parity check bit>,<timeout time>{CR}{LF}

For example: Host sends: AT+UART?{CR}{LF}

Chip responds: 115200,8,1,1,50{CR}{LF}OK{CR}{LF}

Setting command: AT+UART=<baud rate>,<data bit>,<stop bit>,<parity check bit>,<timeout time>{CR}{LF}

Note: This command is used to set the UART parameters, among which the baud rate supports parameters: 9600bit/s, 19200bit/s, 38400bit/s, 57600bit/s, 115200bit/s, and 1000000bit/s. Data bit supports parameters: 8 and 9.Stop bit supports the following parameters: 1 and 2.Parity bit supports the following parameters: 0 (no parity), 1 (odd parity) and 2 (even parity).Timeout time is the data timeout time in transparent transmission

mode, and the unit is mS. After sending this command, the chip will save the set parameters and return the response. After 5mS, the UART will be reinitialized according to the parameters of this configuration.
For example: Host sends: AT+UART=115200,8,1,0,50{CR}{LF} Chip responds: OK{CR}{LF}

12. MAC address

Inquiry command: AT+MAC?{CR}{LF}

Note: This command is used to read the BLE MAC address of the chip, and the returned parameter format is xx:xx:xx:xx:xx:xx. The MAC is in little-endian format, that is, the low byte is first. MAC setting is not supported.

Return: BLE MAC address

For example: Host sends: AT+MAC?{CR}{LF}
Chip responds: 05:DF:39:4C:99:B4{CR}{LF}OK{CR}{LF}

Setting command: AT+MAC=xx:xx:xx:xx:xx:xx{CR}{LF}, the parameter is filled in little-endian format.

Note: MAC parameters are not recommended to be modified. When modified, the chip will not verify the validity of the parameters. This parameter takes effect on the next power up or reset.
For example: Host sends: AT+MAC=05:DF:39:4C:99:B4{CR}{LF} Chip responds: OK{CR}{LF}

13. Transmission power

Inquiry command: AT+TPL?{CR}{LF}

Note: Inquire the current BLE transmission power.

Return: Power level.

For example: Host sends: AT+TPL?{CR}{LF} Chip responds: 0{CR}{LF}OK{CR}{LF}

Setting command: AT+TPL=<x>{CR}{LF}

Note: Set the BLE transmission power, x supports the following parameters: 0 (0DB), 1 (1DB), 2 (2DB), 3 (3DB), 4 (-3DB), 5 (-8DB), 6 (-14DB), 7 (-20DB), other parameters are not supported, the set parameters will be saved and will take effect at the next restart.
For example: Host sends: AT+TPL=1{CR}{LF} Chip responds: OK{CR}{LF}

14. Inquire BLE status

Command: AT+BLESTA?{CR}{LF}

Note: This command inquires the current BLE status. The status is composed of two ASCIIs, which means a HEX. Note to distinguish modes. The following figure shows the status and meaning of each mode.

Broadcast Mode		Host Mode		Slave Mode	
Status Code	Meaning	Status Code	Meaning	Status Code	Meaning
00	Not Initialized	00	Not Initialized	00	Not Initialized
01	Device initialization completed	01	Scan	01	Device initialization completed
02	Broadcast	02	Connecting	02	Broadcast
		03	Connect Successfully	03	Ready broadcast status
		04	Disconnecting	04	Connect timeout
				05	Connect Successfully

07	Error			07	Error

Return: Status code in current mode.

For example: Host sends: AT+BLESTA?{CR}{LF} Chip responds: 02{CR}{LF}OK{CR}{LF}

15. Disconnection

Command: AT+DISCONN{CR}{LF}

Note: This command is used to disconnect the current connection. The status returned by the command is not disconnection but the command is received and can be executed. The disconnection result can be viewed by using the query status.

Return: Status.

For example: Host sends: AT+DISCONN{CR}{LF} Chip returns: OK{CR}{LF}

16. BLE working mode

Inquiry command: AT+BLEMODE?{CR}{LF}

Note: This command is used to return the current BLE working mode. The mode parameters are: 0 (broadcast mode), 1 (host mode), 2 (device mode).

Return: Working mode.

For example: Host sends: AT+BLEMODE?{CR}{LF} Chip responds: 2{CR}{LF}OK{CR}{LF}

Setting command: AT+BLEMODE=<x>{CR}{LF}

Note: The mode setting parameter x can be 0 (broadcast mode), or 1 (host mode) or 2 (device mode). After the command is received, the configuration parameter is saved, and it is not executed until the next restart.

For example: Host sends: AT+BLEMODE=1{CR}{LF} Chip responds: OK{CR}{LF}

17. Get current connected MAC address

Command: AT+CCADD?{CR}{LF}

Note: This command is used to get the MAC address that is currently connected to the chip. The MAC is in little-endian format, that is, low byte first. If there is no connection, empty MAC address is returned.

Return: MAC address connected.

For example: Host sends: AT+CCADD?{CR}{LF}

Chip responds: 05:DF:39:4C:99:B4{CR}{LF}OK{CR}{LF}

18. Chip name

Getting command: AT+NAME?{CR}{LF}

Note: This command is used to get the chip name of current device, in string format.

Return: Current chip name.

For example: Host sends: AT+NAME?{CR}{LF}

Chip responds: WCH MODULE{CR}{LF}OK{CR}{LF}

Setting command: AT+NAME=<chip name string>{CR}{LF}

Note: The length of chip name string does not exceed 18 characters. After the setting command is sent, the chip will keep this parameter and enable it after the next restart. The chip name will be reflected in the broadcast packet in device mode, and the name will be displayed when the host searches. In other modes, this parameter is not used, but the chip name can also be read by using commands.

Return: Status.

For example: Host sends: AT+NAME=TEST NAME{CR}{LF} Chip responds: OK{CR}{LF}

19. Device name

Getting command: AT+PNAME?{CR}{LF}

Note: This command is used to get the device name in device mode and used for attribute description. For

details, please refer to the BLE protocol.

Return: Device name.

For example: Host sends: AT+PNAME?{CR}{LF}

Chip responds: WCH PREI{CR}{LF}OK{CR}{LF}

Configuration command: AT+PNAME=<device name string>{CR}{LF}

Note: This command is used to set the device name. The parameter is saved, and used at the next startup. The length of the name does not exceed 18 characters.

For example: Host sends: AT+PNAME=TEST PNAME{CR}{LF} Chip responds: OK{CR}{LF}

20. Password enable

Getting command: AT+PASEN?{CR}{LF}

Note: This command is used to inquire the current password enabling status, and the returned parameter is "ON" or "OFF".

Return: Password enabling status.

For example: Host sends: AT+PASEN?{CR}{LF} Chip responds: ON{CR}{LF}OK{CR}{LF}

Configuration command: AT+PASEN=<enabling status>{CR}{LF}

Note: The enabling state is also "ON" or "OFF", indicating whether a password is used. It is valid for the slave. When the password is not enabled, a default password value of 0 will be given, and the command will take effect at the next startup.

Return: Execution status.

For example: Host sends: AT+PASEN=ON{CR}{LF} Chip responds: OK{CR}{LF}

21. Password setting

Getting command: AT+PASS?{CR}{LF}

Note: The command is used to inquire the password and return the currently configured slave password.

Return: Slave password

For example: Host sends: AT+PASS?{CR}{LF} Chip responds: 123456{CR}{LF}OK{CR}{LF}

Setting command: AT+PASS=<ASICC value of password>{CR}{LF}

Note: This command is used to configure the connection password of current slave in ASCII format. The parameter set by the command will take effect at the next startup.

Return: Command execution status.

For example: Host sends: AT+PASS=111222{CR}{LF} Chip responds: OK{CR}{LF}

22. System ID of device information

Getting command: AT+SYSID?{CR}{LF}

Note: This command is used to get the system ID of the device information. The system ID is used for the information in device mode. For details, please refer to the BLE protocol. If the user does not set this parameter, an ID will be generated according to the BLE protocol after the chip initialization, and it will be obtained when the getting command is sent. After the user sets this parameter, no new one will be generated after the chip initialization, but the set parameter will be used.

Return: Current system ID.

For example: Host sends: AT+SYSID?{CR}{LF}

Chip responds: 05DF3900004C99B4{CR}{LF}OK{CR}{LF}

Configuration command: AT+SYSID=<system ID>{CR}{LF}

Note: The size of system ID is 8 bytes, that is, 16 characters are sent to configure, and every two characters form a byte. The system ID will be used in the next startup after configuration.

Return: Status.

For example: Host sends: AT+SYSID=0102030405060708{CR}{LF} Chip responds: OK{CR}{LF}

23. Chip name of device information

Getting command: AT+MODNAME?{CR}{LF}

Note: This command is used to get the chip name of device information, which uses the device information attribute parameter in the device mode.

Return: Current chip name of device information.

For example: Host sends: AT+MODNAME?{CR}{LF}

Chip responds: WCH BLE{CR}{LF}OK{CR}{LF}

Setting command: AT+MODNAME=<chip name>{CR}{LF}

Note: This command is used to set the chip name of device information, with the length of not exceeding 18 characters.

Return: Status.

For example: Host sends: AT+MODNAME=TEST MODNAME{CR}{LF}

Chip responds: OK{CR}{LF}

24. Serial number of device information

Getting command: AT+SERINUM?{CR}{LF}

Note: This command is used to get the serial number of current device information, in string format.

Return: Serial number

For example: Host sends: AT+SERINUM?{CR}{LF}

Chip responds: 170621000000{CR}{LF}OK{CR}{LF}

Setting command: AT+SERINUM=<serial number>{CR}{LF}

Note: The serial number is in string format, with the length of not exceeding 18 characters. The parameter is saved, and used at the next startup.

Return: Status.

For example: Host sends: AT+SERINUM=123456{CR}{LF} Chip responds: OK{CR}{LF}

25. Firmware version of device information

Getting command: AT+FIRMREV?{CR}{LF}

Note: The command is used to get the firmware version of device information, in string format.

Return: Firmware version of device information.

For example: Host sends: AT+FIRMREV?{CR}{LF}

Chip responds: VER1.0{CR}{LF}OK{CR}{LF}

Setting command: AT+FIRMREV=<firmware version>{CR}{LF}

Note: The firmware version is in string format, with the length of not exceeding 18 characters. The parameter is saved, and used at the next startup.

Return: Status.

For example: Host sends: AT+FIRMREV=VER1.2{CR}{LF} Chip responds: OK{CR}{LF}

26. Hardware version of device information

Getting command: AT+HARDREV?{CR}{LF}

Note: The command is used to get the hardware version of device information, in string format.

Return: Hardware version of device information.

For example: Host sends: AT+HARDREV?{CR}{LF} Chip returns: VER1.0{CR}{LF}OK{CR}{LF}

Setting command: AT+HARDREV=<hardware version>{CR}{LF}

Note: This command is used to set the hardware version information of the device, in string format, with the length of not exceeding 18 characters.

Return: Status.

For example: Host sends: AT+HARDREV=VER1.2{CR}{LF} Chip responds: OK{CR}{LF}

27. Software version of device information

Getting command: AT+SOFTREV?{CR}{LF}

Note: This command is used to get the software version of device information, in string format.

Return: Software version of device information.

For example: Host sends: AT+SOFTREV?{CR}{LF} Chip responds: VER1.0{CR}{LF}OK{CR}{LF}

Setting command: AT+SOFTREV=<software version>{CR}{LF}

Note: This command is used to set the software version of device information, in string format, with the length of not exceeding 18 characters.

Return: Status.

For example: Host sends: AT+SOFTREV=VER1.2{CR}{LF} Chip responds: OK{CR}{LF}

28. Manufacturer name of device information

Getting command: AT+MANUNAME?{CR}{LF}

Note: This command is used to get the manufacturer name of device information, in string format.

Return: Manufacturer name of device information

For example: Host sends: AT+MANUNAME?{CR}{LF}

Chip responds: WCH{CR}{LF}OK{CR}{LF}

Setting command: AT+MANUNAME=<manufacturer name>{CR}{LF}

Note: This command is used to set the manufacturer name of device information, in string format, with the length of not exceeding 18 characters.

Return: Status.

For example: Host sends: AT+MANUNAME=TEST{CR}{LF} Chip responds: OK{CR}{LF}

29. PNP ID of device information

Getting command: AT+PNPID?{CR}{LF}

Note: This command is used to inquire the PNP ID of the device and return a string in ASC format converted by 7bytes.

Return: PNP ID of device information

For example: Host sends: AT+PNPID?{CR}{LF}

Chip responds: 01020304050607{CR}{LF}OK{CR}{LF}

Setting command: AT+PNPID=<PNP ID>{CR}{LF}

Note: PNP ID is a 7-byte string, that is, 14 characters. The PNP ID should refer to the BLE protocol.

Return: Status.

For example: Host sends: AT+PNPID=10203040506070{CR}{LF} Chip responds: OK{CR}{LF}

30. Broadcast enabling control

Getting command: AT+ADVEN?{CR}{LF}

Note: This command is used to get the current broadcast status.

Return: ON or OFF.

For example: Host sends: AT+ADVEN?{CR}{LF} Chip responds: ON{CR}{LF}OK{CR}{LF}

Setting command: AT+ADVEN=<control>{CR}{LF}

Note: The control parameter set is "ON" or "OFF". When the broadcast data setting is received in the OFF state, the broadcasting function will be automatically enabled.

Return: Status.

For example: Host sends: AT+ADVEN=OFF{CR}{LF} Chip responds: OK{CR}{LF}

31. Broadcast data

Getting command: AT+ADVDAT?{CR}{LF}

Note: This command is used to inquire the current broadcast data.

Return: Broadcast data.

For example: Host sends: AT+ADVDAT?{CR}{LF} Chip responds: 02010604FF010203{CR}{LF}

Setting command: AT+ADVDAT=<broadcast data>{CR}{LF}

Note: This command is used to set the broadcast data, maximum 31 bytes, and the transmission format is ASC format. The broadcast data format should refer to the BLE protocol.

Return: Status.

For example: Host sends: AT+ADVDAT=02010604FF010203{CR}{LF}

Chip responds: OK{CR}{LF}

32. Serial number connection

Command: AT+LINK=<serial number>,<password>{CR}{LF}

Note: After scanning, specify a serial number connection starting from 1 based on the serial number given by the scan.

Return: Command operation status.

For example: Host sends: AT+LINK=1,123456{CR}{LF} Chip responds: OK{CR}{LF}

33. Direct connection

Command: AT+CONN=<MAC>,<password>{CR}{LF}

Note: This command is used to directly connect the BLE device with known MAC address, and the MAC data format is xx:xx:xx:xx:xx:xx. The MAC is in little-endian format, that is lower bytes first, and the password is in string format.

Return: Command operation result.

For example: Host sends: AT+CONN=05:DF:39:4C:99:B4,123456{CR}{LF}

Chip responds: OK{CR}{LF}

34. Host scan command

Command: AT+SCAN=<control>{CR}{LF}

Note: Control scan, the parameter is "ON" or "OFF". It is recommended to enable the scan, and wait until the scan ends automatically. At this time, the UART will output the scan result and the serial number corresponding to the device, and it is not necessary to send a stop command.

Return: Command operation result.

For example: Host sends: AT+SCAN=ON{CR}{LF} Chip responds: OK{CR}{LF}

35. Default connection parameters

Getting command: AT+CONADD?{CR}{LF}

Note: This command is used to inquire the internally saved default connection parameter. If the parameter is valid, it will be automatically connected after the chip is initialized.

Return: Internally saved default connection parameters.

For example: Host sends: AT+CONADD?{CR}{LF}

Chip responds: 05:DF:39:4C:99:B4,123456{CR}{LF}

Setting command: AT+CONADD=<MAC>,<password>{CR}{LF}

Note: This command is used to set the default connection parameter. The MAC is in little-endian format, that is, low bytes first.

Return: Command execution result.

For example: Host sends: AT+CONADD=05:DF:39:4C:99:B4,123456{CR}{LF}

Chip responds: OK{CR}{LF}

36. Clear default connection parameters

Command: AT+CLRCONADD{CR}{LF}

Note: The command is used to clear the default connection parameter and save the result.

Return: Command execution status.

For example: Host sends: AT+CLRCONADD{CR}{LF} Chip responds: OK{CR}{LF}

37. Read RSSI

Command: AT+RSSI=<control>,<parameter>{CR}{LF}

Note: Reading RSSI can only be effective after connection, and the command may be executed. But when BLE is not connected, the RSSI will not be read, nor will it return RSSI data. This command can only be used to set but not query, where the control is "ON" or "OFF". No parameter is required for OFF. The parameter is the cycle of reading the RSSI in unit of millisecond. When the parameter is 0, it means a single read, and the time interval of single read is 50mS. After the chip responds, it will return the read value of RSSI according to the set loop time length. The returned value will always be returned in AT mode. It is recommended that the ON or OFF is executed in the AT state to prevent conflicts with the data part. The time interval should not be less than 50mS, so as to ensure that the UART can output normally, and there is an interval for BLE processing.

Return: Command execution status.

For example: Host sends: AT+RSSI=ON,1000{CR}{LF} Chip responds: OK{CR}{LF}

For example: Host sends: AT+RSSI=OFF{CR}{LF} Chip responds: OK{CR}{LF}

38. Read ADC value

Command: AT+ADC?{CR}{LF}

Note: This command is used to read the ADC sample value, which will be returned in ASCII format. The effective range of the returned value is 0-4095, that is, the range of the 12-bit ADC sample value.

Return: ADC sample value.

For example: Host sends: AT+ADC?{CR}{LF} Chip responds: 2048{CR}{LF}OK{CR}{LF}

39. Set chip sleep mode

Inquiry command: AT+SLEEP?{CR}{LF}

Note: This command is used to inquire the current sleep mode.

Return: Current sleep mode.

For example: Host sends: AT+SLEEP?{CR}{LF} Chip responds: 1{CR}{LF}OK{CR}{LF}

Setting command: AT+SLEEP=<x>{CR}{LF}

Note: This command is used to set the chip to enter low energy mode. And x supports the following parameters: 0 (not enter the low energy mode), 1 (sleep mode), 2 (power-down mode). Other parameters are not supported. Setting parameters will be saved and take effect in the next restart.

For example: Host sends: AT+SLEEP=1{CR}{LF} Chip responds: OK{CR}{LF}

40. Read supply voltage of chip

Command: AT+BAT?{CR}{LF}

Note: This command is used to read the supply voltage of the current chip, that is, the VCC voltage value.

Return: A value in unit of mV.

For example: Host sends: AT+BAT?{CR}{LF} Chip responds: 2985{CR}{LF}OK{CR}{LF}

41. Host scan to show slave voltage

Inquiry command: AT+BDSP?{CR}{LF}

Note: This command is used to inquire the setting status of host scan to show slave voltage.

Return: Current setting status.

For example: Host sends: AT+BDSP?{CR}{LF} Chip responds: ON{CR}{LF}OK{CR}{LF}

Setting command: AT+BDSP=<control>{CR}{LF}

Note: The value of <control> is "ON" or "OFF", which corresponds to turning on and off the voltage echo. This parameter will affect displayed information of slave when the host is scanning. For details, refer to the scanning introduction in the description of the host mode below.

For example: Host sends: AT+BDSP=ON{CR}{LF} Chip returns: OK{CR}{LF}

42. BLE configuration interface control

Inquiry command: AT+BLECFGEN?{CR}{LF}

Note: Inquire the current BLE configuration interface switch status. This state means that acquisition and configuration of chip parameter through BLE interface are in enabled state. When the state is OFF, the BLE interface cannot acquire and configure parameters until it is reopened by using the AT command.

Return: 0{CR}{LF}OK{CR}{LF} or 1{CR}{LF}OK{CR}{LF} corresponds to turn off the BLE configuration interface and turn on the BLE control interface respectively.

For example: Host sends: AT+BLECFGEN?{CR}{LF} Chip responds: 1{CR}{LF}OK{CR}{LF}

Setting command: AT+BLECFGEN=<parameter>{CR}{LF}

Note: Same as the inquiry command above, the parameter is 0 or 1, corresponding to turn off and turn on respectively.

For example: Host sends: AT+BLECFGEN=1{CR}{LF} Chip returns: OK{CR}{LF}

43. Set broadcast channel

Inquiry command: AT+BCCH?{CR}{LF}

Note: This command is used to inquire the current broadcast channel parameters and return the following values: 0 corresponding to enabling three channels, 1 corresponding to broadcast in channel 37 only, 2 corresponding to broadcast in channel 38 only, and 3 corresponding to broadcast in channel 39 only. Channels 37, 38 and 39 are defined by the BLE protocol. The set parameters take effect after restart.

For example: Host sends: AT+BCCH?{CR}{LF} Chip responds: 0{CR}{LF}OK{CR}{LF}

Setting command: AT+BCCH=<parameter>{CR}{LF}

Note: This command is used to set the broadcast channel parameter, which affects the broadcast channel settings in broadcast mode and slave mode.

For example: Host sends: AT+BCCH=0{CR}{LF} Chip responds: OK{CR}{LF}

44. Set broadcast interval

Inquiry command: AT+ADVINTER?{CR}{LF}

Note: This command is used to inquire the current broadcast interval and return a value in unit of 0.625mS. The default value is 160, which is a broadcast interval of 100mS.

For example: Host sends: AT+ADVINTER?{CR}{LF} Chip responds: 160{CR}{LF}OK{CR}{LF}

Setting command: AT+ADVINTER=<parameter>{CR}{LF}

Note: This command is used to set the broadcast interval and parameter is a value.

For example: Host sends: AT+ADVINTER=1600{CR}{LF} Chip responds: OK{CR}{LF}

45. Set connection interval AT+CONNINTER

Inquiry command: AT+CONNINTER?{CR}{LF}

Note: This command is used to inquire the connection interval parameters of current chip and return a range value corresponding to the minimum value and the maximum value respectively. During connection, the BLE protocol will negotiate a communication time. If the parameter does not meet the requirement, the chip

will initiate a renegotiation. The unit of value is 1.25mS.

For example: Host sends: AT+CONNINTER?{CR}{LF} Chip responds: 6-16{CR}{LF}OK{CR}{LF}

Setting command: AT+CONNINTER=<parameter 1>-<parameter 2>{CR}{LF}

Note: This command is used to set the connection interval parameter, which should meet the requirements of BLE protocol, and the set value should not exceed 65535.

For example: Host sends: AT+CONNINTER=6-160{CR}{LF} Chip responds: OK{CR}{LF}

46. Set internal 32K clock calibration AT+LSICALI

Inquiry command: AT+LSICALI?{CR}{LF}

Note: The chip returns the current internal 32K clock calibration parameter with three parameter fields separated by commas. The parameter format is defined as: <calibration mode>,<temperature difference threshold>,<timing time>{CR}{LF}. The calibration mode value is defined as: 0: close calibration, 1: calibrate with timing mode, 2: calibrate with temperature difference. Temperature difference threshold: The set value is the difference of temperature difference change, which will be used when calibrating with temperature difference after set. Timing time: Set two calibration methods to time in unit of mS.

For example: Host sends: AT+LSICALI?{CR}{LF} Chip responds: 2,7,5000{CR}{LF}OK{CR}{LF}

Setting command: AT+LSICALI=<calibration mode>,<temperature difference threshold>,<timing time>{CR}{LF}

Note: This command is used to set the calibration mode of current LSI clock. For definition of parameters, refer to the inquiry command.

For example: Host sends: AT+LSICALI=2,10,10000{CR}{LF} Chip responds: OK{CR}{LF}

47. Set RF calibration AT+RFCALI

Inquiry command: AT+RFCALI?{CR}{LF}

Note: The chip returns the current RF calibration parameters with three parameter fields separated by commas. The parameter format is defined as: <calibration mode>,<temperature difference threshold>,<timing time>{CR}{LF}. The calibration mode value is defined as: 0: close calibration, 1: calibrate with timing mode, 2: calibrate with temperature difference. Temperature difference threshold: The set value is the difference of temperature difference change, which will be used when calibrating with temperature difference after set. Timing time: Set two calibration methods to time in unit of mS. The formats of the two calibration setting commands are the same.

For example: Host sends: AT+RFCALI?{CR}{LF} Chip responds: 2,7,5000{CR}{LF}OK{CR}{LF}

Setting command: AT+RFCALI=<calibration mode>,<temperature difference threshold>,<timing time>{CR}{LF}

Note: This command is used to set the current RF calibration mode. For definition of parameters, refer to the inquiry command.

For example: Host sends: AT+RFCALI=2,10,10000{CR}{LF} Chip responds: OK{CR}{LF}

48. Set TNOW pin function AT+TNOW

Inquiry command: AT+TNOW?{CR}{LF}

Note: The chip returns the current TNOW pin parameters with two parameter fields separated by commas. The parameter format is defined as: <TNOW enabling>,<polarity selection>{CR}{LF}. The enabling parameter value is defined as: 0: Disable TNOW pin function; 1: Enable TNOW pin output. The polarity selection parameter value is defined as: 0: Output in default mode, 1: Output in reverse in default mode. The format of setting parameter is same as that of inquiry parameter.

For example: Host sends: AT+TNOW?{CR}{LF} Chip responds: 1,0{CR}{LF}OK{CR}{LF}

Setting command: AT+TNOW=<TNOW enabling>,<polarity selection>{CR}{LF}

Note: This command is used to set the TNOW pin function. For definition of parameters, refer to the inquiry command.

For example: Host sends: AT+TNOW=1,0{CR}{LF} Chip responds: OK{CR}{LF}

49. Set BLESTA pin function AT+BSTA

Inquiry command: AT+BSTA?{CR}{LF}

Note: The chip returns the current BLESTA pin parameter with two parameter fields separated by commas. The parameter format is defined as: <BLESTA pin enabling>,<polarity selection>{CR}{LF}. The enabling parameter value is defined as: 0: Disable BLESTA pin function; 1: Enable BLESTA pin output. The polarity selection parameter value is defined as: 0: Output in default mode, 1: Output in reverse in default mode. The format of setting parameter is the same as that of inquiry parameter.

For example: Host sends: AT+BSTA?{CR}{LF} Chip responds: 1,0{CR}{LF}OK{CR}{LF}

Setting command: AT+BSTA=<BLESTA pin enabling>,<polarity selection>{CR}{LF}

Note: This command is used to set the BLESTA pin function. For definition of parameters, refer to the inquiry command.

For example: Host sends: AT+BSTA=1,0{CR}{LF} Chip responds: OK{CR}{LF}

50. Set flow control function AT+AFEC

Inquiry command: AT+AFEC?{CR}{LF}

Note: The chip returns the current flow control settings and the output status of DTR and RTS pins with three parameter fields separated by commas. The parameter format is defined as: <flow control status>,<DTR pin output level>,<RTS pin output level>{CR}{LF}. The parameter value of flow control status is defined as: 0: Disable serial flow control; 1: Enable serial flow control. The parameter value of DTR pin output level is defined as: 0: output high level, 1: output low level. The parameter value of RTS pin output level is defined as: 0: output high level, 1: output low level. The format of setting parameter is the same as that of inquiry parameter. The flow control setting only affects the RTS, the CTS function is not affected. After setting, the chip will not save it, and the flow control will be turned on by default after power down. When the DTR and RTS are set multiplexed into other pin functions, that will not affect their levels, and the current output status can still be obtained by obtaining the parameters.

For example: Host sends: AT+AFEC?{CR}{LF} Chip responds: 1,1,1{CR}{LF}OK{CR}{LF}

Setting command: AT+AFEC=<flow control status>,<DTR pin output level>,<RTS pin output level>{CR}{LF}

Note: This command is used to set the flow control and related pin functions. For definition of parameters, refer to the inquiry command.

For example: Host sends: AT+AFEC =1,1,1{CR}{LF} Chip responds: OK{CR}{LF}

51. Set GPIO function enabling AT+IOEN

Getting command: AT+IOEN?{CR}{LF}

Note: This command is used to get the GPIO enabling byte in the configuration. Each bit in the HEX format of this value represents the corresponding GPIO ON or OFF. The value obtained is in ASCII format converted from HEX format, such as: 0xFF, which means binary 11111111, bit7-bit0 respectively correspond to GPIO7-GPIO0. GPIO3-GPIO0 means that the synchronous transmission is not affected by this parameter; 0 means that the GPIO function is OFF, and 1 means that the GPIO function is ON.

Return: Execution status.

For example: Host sends: AT+IOEN?{CR}{LF} Chip responds: FF{CR}{LF}OK{CR}{LF}

Setting command: AT+IOEN=<8 IOs enabling>{CR}{LF}

Note: The set parameter format is consistent with the acquired parameter format. Each bit indicates the

corresponding IO enabling status. After setting, the host will save the parameters.

Return: Execution status

For example: Host sends: AT+IOEN=0F{CR}{LF}, the GPIO7-GPIO4 are disabled.

Chip responds: OK{CR}{LF}

7. Operation Instruction

7.1 Broadcast Mode

Step1. Enter AT mode to set basic parameters, baud rate, working mode, etc. Restart the chip and the UART will work according to the set parameters;

Step2. Enter AT mode;

Step3. Send setting broadcast command parameter. It is suggested that the sending interval should be greater than or equal to 100mS. The packet size must not exceed the size of the broadcast packet and the data format of the packet should refer to that of the BLE protocol;

Step4. Monitor broadcast data testing by software.

7.2 Host Mode

Step1. Enter AT mode to set basic parameters, baud rate, working mode, etc. Restart the chip and the UART will work according to the set parameters. You can set the default connection address to make the chip automatically connect to this address, or use the scan connection method;

Step2. Enter AT mode. If the default connection address method is used, the data can be sent after successful connection. You can get the connection status by querying the BLE connection status. Data can be sent only after the connection is successful, otherwise the data will be temporarily stored or discarded.

If scan connection method is used, the scan command will be sent. An example of output data format when scanning is: "1. MAC:4c:df:39:4c:99:b4 RSSI -37dB{CR}{LF}", with the format of "<serial number>"<.><space><MAC address><space><RSSI value>". Every device is searched, one line of output information is occupied, and the ending character string "SCAN END{CR}{LF" }" is output after scanning ends. If the chip turns on the control of the slave supply voltage display, the slave supply voltage will be displayed in the output data, in the format: <serial number><.><space><MAC address><space><RSSI value>< Space><BAT><space><voltage value><mV>". For example: "1. MAC:4c:df:39:4c:99:b4 RSSI -37dB BAT 3298mV{CR}{LF}". At this time, the serial number connection command can be used to select one of the devices for connection. After successful connection and initialization of connection, the chip will output the string "LINK OK{CR}{LF}", indicating that the chip has successfully connected and can exit AT mode for data transparent transmission. When there is a pairing error in the connection, the host will not reconnect to the slave, and the UART will output the string "PAIR ERR{CR}{LF}". At this time, you need to check whether the password parameters are correct.

It should be noted that if the connection is successfully queried by using query BLE connection status, it does not mean that data can be sent immediately. After connection, BLE needs to initialize the connection. At this time, if the data is transparently transmitted, it will be saved, and transmitted after the initialization of the connection or it will be discarded. After connection information is configured and then waiting for string output of successful connection, the method of scanning connection or directly inputting connection address can be used. After scanning ends and then serial number connection is started, the method of scanning connection can be used.

Step3. The chip can exit the AT mode after it is informed that the connection is successful;

Step4. Perform serial port transparent transmission with the slave. Entering AT configuration during the transparent transmission period will cause that data received by BLE is directly discarded.

7.3 Slave Mode

Step1. Set the basic parameters, baud rate, working mode, etc. The chip restarts, the UART will work according to the set parameters, and the relevant setting information parameters of the device mode can be set.

Step2. Enter the AT mode to query the connection status, or directly perform the transparent transmission. When not connected, the transmitted data will be directly discarded;

Step3. Use software or the host to connect to the chip for transparent data transmission. During the transparent transmission period, AT configuration will cause that data received by BLE is directly discarded.

8. BLE Interface Configuration

8.1 Description of BLE Interface Configuration

When CH9141 works in BLE slave mode, it can be configured via BLE. The configuration channel is a custom transmission channel, and the UUID is 0XFFF3. The transmission format is strictly in accordance with the frame transmission that splits a frame into several packets of BLE transmission data during transmission. After reading a frame of data, the BLE can end reading response. The configuration function supports the notification service. When the BLE host opens the notification service, the chip will report the UART transmission status and the MODEM signal status.

8.2 Format Description

The transmission data frame format is shown in the table:

Field Definition	Command code	Data Length	Status Parameter Code	Data Segment	Verify
Field Length	1	2	1	N	1

Command code: 1 byte, indicating the operation of the frame. The range of the command code sent by the host is 0x00-0x7F.The slave responds the command code: the highest bit of the host command code is set to 1 (host command code|0x80);

Data length: 2 bytes, the size is N+2, that is, the total length of the status parameter code, data segment, and verify;

Status parameter code: 1 byte, which serves as the status or parameter information segment when transmitting different command codes;

Data segment: N bytes, the length is the size of the transmitted data, and the value settings in the data are unified in little-endian format;

Verify: 1 byte, the value is the sum of the status parameter code and the data segment.

After the host sends a configuration frame, it must obtain the response frame from the slave, except for the frame without response such as the reset frame.

8.3 Parameter Configuration

8.3.1 Parameter Description

There are two configuration parameters: BLE configuration information and device information in slave mode. The BLE configuration information mainly configures the BLE working mode, UART parameters, version information, etc., and the device information in slave mode mainly configures the device information properties in slave mode, including: System ID, Model Number String, Serial Number String, Firmware Revision String, Hardware Revision String, Software Revision String, Manufacturer Name String, PnP ID.

8.3.2 BLE Configuration Information

Offset Position	Size	Name	Function	Remark
0	1	Parameter flag	Identify parameter validity	Do not modify this value
1	20	Device name	Device name recognized by the host	In the response packet of broadcast packet
21	6	Device MAC address	MAC address of current chip	It is suggested not to modify
27	6	Connection MAC address	Record the other party's MAC after connection	Read Only
33	2	Revision number	Current chip version	Read Only
35	30	Welcome note		Reserved parameter
65	4	Baud rate	Baud rate	
69	1	UART data bit	UART data bit	
70	1	UART parity bit	UART parity bit	
71	1	UART stop bit	UART stop bit	
72	2	UART package timeout	UART timeout time	
74	4	Low energy sleep time	Configure custom sleep time	Reserved parameter
78	1	Low energy mode	Set low-energy mode	
79	1	Chip working mode	Set BLE working mode of chip	
80	1	Chip transmit power	Set BLE transmit power	
81	1	Broadcast enabling	Reserved parameter	Broadcast mode parameter
82	1	Broadcast Mode	Reserved parameter	Broadcast mode parameter
83	2	Broadcast time	Reserved parameter	Broadcast mode parameter
85	2	Minimum connection interval	Set the default minimum connection parameter	Salve mode parameter
87	2	Maximum connection interval	Set the default maximum connection parameter	Salve mode parameter
89	2	Timeout time	Set the BLE connection timeout parameter	Salve mode parameter
91	20	Device name	Display in DeviceName of Generic Access	Salve mode parameter

111	1	Password enabling		Salve mode parameter
112	1	Password length		Salve mode parameter
113	6	Password		Salve mode parameter
119	4	Default connection MAC flag	Save 4 sets of default connection parameters by default	Host mode parameter
123	24	Default connection MAC	4*6: 4 sets of default connection MAC parameter	Host mode parameter
147	24	Password of default connection MAC	4*6: 4 sets of default connection MAC password parameter	Host mode parameter
171	1	GPIO mode setting	Set GPIO 1: output; 0: input	Each bit corresponds to one IO bit7-0 correspond to GPIO7-0
172	1	GPIO level value setting	Set GPIO output level 1: output high level; 0: output low level	Each bit corresponds to one IO bit7-0 correspond to GPIO7-0

8.3.3 Slave Device Information

Offset Position	Size	Name	Function	Remark
0	1	Parameter flag	Identify parameter validity	Do not modify this value
1	8	System ID	System ID	
9	20	Model Number String	Device number	String format
29	20	Serial Number String	Serial number	String format
49	20	Firmware Revision String	Firmware version number	String format
69	20	Hardware Revision String	Hardware version number	String format
89	20	Software Revision String	Software version number	String format
109	20	Manufacturer Name String	Manufacturer name	String format
129	7	PnP ID	PnP ID	

8.3.4 BLE Control Parameter

This parameter is supported after version v1.04. The parameter contains a "block" with a size of 512 bytes. Refer to the table below for the range of parameters currently used. Other unused parts are reserved for future use.

Offset Position	Size	Name	Function	Remark
0	1	IMAGE information	Current image information	Do not modify this value
1	3	Reserved parameter		Do not modify this value
4	1	Parameter flag	Identify parameter validity	Do not modify this value
5	2	Version information	Current chip version	Do not modify this value
7	1	BLE configuration interface enabling	Enable/disable BLE configuration interface	Set the parameter to 0x00 or 0x01, corresponding to OFF or ON. If set to OFF, the BLE will have no access to the chip configuration
8	1	Broadcast channel configuration	Configure broadcast channel parameter	The value can be set to 0x07, 0x01, 0x02 or 0x04, corresponding to broadcast in three broadcast channels, broadcast in channel 37 only, broadcast in channel 38 only, and broadcast in channel 39 only
9	1	Control configuration, define switches by bit	bit7: Parameter validity	Do not modify this value
			bit6: Display slave supply voltage while scanning	0: OFF 1: ON
10	2	Reserve control configuration		Do not modify this value
12	1	Internal LSI calibration method	0: Close calibration 1: Adopt timing method 2: Adopt temperature difference method	Added for v1.06
13	1	BLE RF calibration method	0: Close calibration 1: Adopt timing method 2: Adopt temperature difference method	
14	1	Temperature change threshold of internal LSI calibration	Temperature difference value	
15	1	Temperature change threshold of BLE RF calibration	Temperature difference value	
16	4	Timing time of internal LSI calibration	Timing time in unit of mS	
20	4	Timing time of BLE RF calibration	Timing time in unit of mS	
21	1	GPIO function enable	Define GPIO enable by bit	Added for v1.07

22	1	TNOW pin enable	bit4: Polarity control bit0: Enable	
23	1	BLESTA pin enable	bit4: Polarity control bit0: Enable	
24	1	Reserved parameter		
25-511		Reserved parameter		Do not modify this value

8.4 Configuration Command Description

8.4.1 Get Configuration

The host sends and gets the configuration frame in the following format:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function
0x01	0x00 0x02	0x01	Blank	0x01	Get BLE configuration information
0x01	0x00 0x02	0x02	Blank	0x02	Get device information in slave mode
0x01	0x00 0x02	0x03	Blank	0x02	Get BLE control parameter

The chip responds:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function
0x81	Configuration information length +2	0x01	Configuration information	check	Return BLE configuration information
0x81	Device information length +2	0x02	Device information	check	Return device information in slave mode
0x81	BLE control parameter length +2	0x03	BLE control parameter	check	Return BLE control parameter

8.4.2 Configuration Parameter

The host sends the configuration parameter frame in the following format:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function
0x02	Configuration information length +2	0x01	Configuration information	check	Configure the BLE configuration parameter
0x02	Device information length +2	0x02	Device information	check	Configure the device information in slave mode
0x02	BLE control parameter length +2	0x03	BLE control parameter	check	Configure the BLE control parameter

The chip responds:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function
0x82	0x00 0x02	0x01	Blank	0x01	Configure parameter response
0x82	0x00 0x02	0x02	Blank	0x02	Configure device information response
0x82	0x00 0x02	0x03	Blank	0x03	Configure the BLE control parameter response

8.4.3 Reset Parameter

The host sends the reset parameter frame in the following format:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function
0x03	0x00 0x02	0x01	Blank	0x01	Reset the BLE configuration parameter
0x03	0x00 0x02	0x02	Blank	0x02	Reset device information in slave mode
0x03	0x00 0x02	0x03	Blank	0x02	Reset the BLE control parameter

The chip responds:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function
0x83	0x00 0x02	0x01	Blank	0x01	Reset parameter response
0x83	0x00 0x02	0x02	Blank	0x02	Reset device information response
0x83	0x00 0x02	0x03	Blank	0x03	Reset the BLE control parameter response

8.4.4 Reset Chip

The host sends the reset parameter frame in the following format:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function
0x04	0x00 0x02	0x00	Blank	0x00	Reset BLE configuration information

The chip will reset directly when the reset frame is received without responding it.

8.4.5 GPIO Related Commands

The host sends the GPIO configuration parameter frame in the following format:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function

0x05	0x00 0x04	0x01	NUM+DIR	check	Set the direction of NUM NUM: 0x04 - 0x07 DIR: 0x00 input, 0x01 output
0x05	0x00 0x03	0x02	NUM	check	Read the level status of NUM GPIO
0x05	0x00 0x04	0x03	NUM+VAL	check	Set the level value of NUM GPIO VAL: 0x00 low level, 0x01 high level
0x05	0x00 0x04	0x04	GPIO1+GPIO3	check	Synchronization parameter, generally used to connect two chips, other hosts do not need to operate
0x05	0x00 0x02	0x05	Blank	0x05	Read ADC value

The chip responds:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function
0x85	0x00 0x03	0x01	STA	check	Return the status of the set GPIO, 0 means success, and others mean failure
0x85	0x00 0x04	0x02	NUM+VAL	check	Read the level status of the read GPIO
0x85	0x00 0x03	0x03	STA	check	Return the status of the set level value 0 means success, and others mean failure
0x85	0x00 0x04	0x04	GPIO1+GPIO3	check	Synchronization parameter, generally used to connect two chips, other hosts do not need to operate
0x85	0x00 0x04	0x05	ADC_VAL	check	Return the read ADC value, 2 bytes

8.4.6 UART Parameter Setting

The host sends the UART configuration parameter frame in the following format:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function
0x06	0x00 0x09	0x00	UART parameters	check	Configure UART parameters

The format of UART parameters is defined as:

Offset position	Size	Name	Function	Remark
0	4	Baud rate		Little-endian format
4	1	Data bit	UART data length	Parameter value: 5-8
5	1	Stop bit	Stop bit format setting	Parameter value: 1-2

6	1	Parity check bit	Parity check bit format selection	0: No check 1: Odd parity 2: Even parity 3: Flag bit 4: blank bit

The chip responds:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function
0x86	0x00 0x09	0x00	UART parameters	check	The content of the response parameter is consistent with that set

8.4.7 Flow Control Function Setting

The host sends the flow control setting frame in the following format:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function
0x07	0x00 0x05	0x00	Flow control function parameter	check	Configure flow control function

The format of flow control parameter is defined as:

Offset position	Size	Name	Function	Remark
0	1	Flow control function enable		0: Disable 1: Enable
1	1	DTR pin setting	DTR output level setting	0: DTR pin output high 1: DTR pin output low
2	1	RTS pin setting	RTS output level setting	0: RTS pin output high 1: RTS pin output low

The chip responds:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function
0x87	0x00 0x05	0x00	Flow control function parameter	check	The content of the response parameter is consistent with that set

8.4.8 UART and MODEM Status Report

The chip responds:

Command code	Data Length	Status Parameter Code	Data Segment	Check	Function

0x88	0x00 0x04	0x00	UART status parameter	check	UART and MODEM status report

The format of UART status parameter is defined as:

Offset position	Size	Name	Function	Remark
0	1	UART status	bit1: MODEM change bit0: UART sending blank	Definition of corresponding bit value: 1: Status valid 0: Status invalid
1	1	MODEM status	bit7: The bit of the DCD pin is reversed, valid as 1 bit6: The bit of the RI pin is reversed, valid as 1 bit5: The bit of the DSR pin is reversed, valid as 1 bit4: The bit of the CTS pin is reversed, valid as 1 bit3: A value of 1 indicates a change in DCD pin bit2: A value of 1 indicates a change in RI pin bit1: A value of 1 indicates a change in DSR pin bit0: A value of 1 indicates a change in CTS pin	

After the configuration channel notification property is enabled, CH9141 detects the change of UART status and sends the frame to the host via BLE.

9. Parameters

9.1 Absolute Maximum Value

Critical value or exceeding the absolute maximum value may cause the chip to work abnormally or even be damaged.

Name	Parameter description	Min.	Max.	Unit
TA	Ambient temperature during operation	-40	85	℃
TS	Ambient temperature during storage	-40	105	℃
VCC	System supply voltage	-0.4	3.9	V
VIO	Voltage of the input or output pins	-0.4	VCC+0.4	V

9.2 Electrical Parameters

Name	Parameter description	Min.	Typ.	Max.	Unit
VCC	Supply voltage	2.5	3.3	3.6	V
VIL	Low level input voltage	0		0.9	V
VIH	High level input voltage	2.0		VCC	V
VOL	Low level output voltage	0	0.3	0.4	V
VOH	High level output voltage	VCC-0.4	VCC-0.3	VCC	V
IUP	Input current of the input terminal with built-in pull-up resistor	25	60	90	uA